CN AF/2132

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of: Alan FOLMSBEE	Group Art Unit: 2132
	Examiner: LANIER, B. E.
Application No: 09/376,654	Atty. Docket No: SUNMP210 (Previously, 5437-076/P41)
Filed: August 18, 1999	Date: May 11, 2005
For: SECURE PROGRAM EXECUTION DEPENDING ON PREDICTABLE ERROR CORRECTION	

CERTIFICATE OF MAILING
I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner for Patents, Mail Stop Appeal Brief-Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on **May 11, 2005.**

Signed: Cynthia Dawn
Cynthia Dawn

TRANSMITTAL OF APPEAL BRIEF
(PATENT APPLICATION -- 37 CFR 192)

Commissioner for Patents
Mail Stop: Appeal Brief-Patents
Alexandria, VA 22313-1450

Sir:

This Appeal Brief is in furtherance of the Notice of Appeal filed in this case on March 11, 2005. The Notice of Appeal was received by the USPTO on April 29, 2004. Therefore, the due date for this Appeal Brief is May 11, 2005, with a two-month extension of time:

This application is on behalf of:

☐ Small Entity ☒ Large Entity

Pursuant to 37 CFR 1.17(f), the fee for filing the Appeal Brief is:

☐ $250.00 (Small Entity) ☒ $500.00 (Large Entity)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136 apply:

☐ Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d)) for the total number of months checked below:

Months	Large Entity	Small Entity
☐ one	$120.00	$60.00
☐ two	$450.00	$225.00
☐ three	$1,020.00	$510.00

☒ If an additional extension of time is required, please consider this a petition therefor.

☐ An extension for __ months has already been secured and the fee paid therefor of $ is deducted from the total fee due for the total months of extension now requested.

☒ Applicant believes that no extension of term is required. However, this conditional petition is being made to provide for the possibility that Applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Total Fees Due:

Appeal Brief Filing Fee	$500.00
Extension Fee (if any)	$0.00
Total Fee Due	**$500.00**

☒ Enclosed is Check No. 14090 in the amount of $500.00.

☒ Charge any additional fees or credit any overpayment to Deposit Account No. 50-0850, (Order No. SUNMP210). One copy of this transmittal is enclosed.

Respectfully submitted,
MARTINE PENILLA & GENCARELLA, LLP

Leonard Heyman
Registration No. 40,418

710 Lakeway Drive, Suite 200
Sunnyvale, CA 94085
Tel.: (408) 774-6933
Fax: (408) 749-6901
Customer No. 32291

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application for Patent

FOR:

SECURE PROGRAM EXECUTION DEPENDING ON PREDICTABLE ERROR CORRECTION

APPEAL BRIEF

EX PARTE Alan C. Folmsbee

Application No. 09/376,654
Filed August 18, 1999
Technology Center/Art Unit 2132

Submitted in accordance with 37 C.F.R. §41.67

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on May 11, 2005.

Signed: Cynthia Dawn
Cynthia Dawn

MARTINE PENILLA & GENCARELLA, LLP
Attorneys for Appellant

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest is Sun Microsystems, Inc., the assignee of the present application.

II. RELATED APPEALS AND INTERFERENCES

The undersigned is not aware of any related appeals or interferences.

III. STATUS OF THE CLAIMS

Claims 1, 3, 4, 13, 17, and 18 are pending in the subject application. Claims 1, 3, 4, 13, 17, and 18 have been finally rejected and are on appeal.

IV. STATUS OF THE AMENDMENTS

No amendments, cancellations or other changes have been made after the Final Office Action of December 15, 2004.

V. SUMMARY OF THE INVENTION

The present invention relates to a microprocessor for executing computer instructions. The microprocessor presently disclosed is modified from a typical microprocessor in that it includes a number of security enhancements. The security enhancements allow the microprocessor to execute code that will not execute on other processors. Thus, the software must be specifically modified for the particular processor and, once modified, will not run on other processors. Since the software will not run on other processors the problem of software piracy is effectively addressed. The processor is further designed to obfuscate the specific modifications and make reverse engineering difficult.

The present Application discloses a number of features of a microprocessor that relate to various techniques to thwart software piracy. However, the appealed claims focus on only one feature, which is set forth in the specification beginning at line 7 of page 22 and is discussed with respect to Figure 11.

Referring to Figure 11, a central processor unit (CPU) 161 receives error correction information, a key, and instructions. The key and error correction information are used to

process instructions that have errors intentionally introduced. The instructions are processed using error correction circuitry 169 to produce corrected instructions 173.

The present application uses the word "encryption" to include intentionally placing errors into data and/or instructions, which are then corrected prior to execution (page 22, line 8-9). These errors are then corrected using on-chip circuitry (page 22, line 10). An exemplary error correcting algorithm that may be used to correct the intentionally inserted errors is the well-known Hamming code (page 22, line 13-14). Developed in 1950, the Hamming code and derivatives thereof have been used, for example, in correcting soft errors, which are errors in transmission of information and errors in computer memory caused by ionizing radiation, such as neutron radiation from space. The present Application discusses error correction according to a selected error correction scheme for the purpose of correcting errors intentionally inserted into the instructions.

One aspect of the invention is that the error correction circuitry is on the same processor that executes them (page 22, lines 21-22) rather than on an adjacent processor or error correction through software means. Thus, there is no error-corrected instruction stream that is readily accessible. As mentioned in the written description, "in the past, secured embedded microcontrollers have usually used standard microprocessor architectures and have attached security enhancing hardware around this core. In the inventive configuration, the architecture is designed with physical security in mind so that the security hardware features are deeply embedded in the architecture, instead of around the periphery of the architecture" (page 31, lines 1-5).

VI. ISSUES

There is one issue on appeal: whether claims 1, 3, 4, 13, 17, and 18 is anticipated by U.S. Patent 4,864,949 issued to Kobus (Kobus).

VII. ARGUMENTS

In the discussion below, all references to "the Office Action" will be made, unless otherwise noted, to the Final Office Action mailed December 15, 2004. All references to "the

Advisory Action," will be referencing the Advisory Action mailed February 15, 2005. All references to the claims and to specific lines in the claims will be made in reference to the claims in the attached claim listing (section VIII). When line numbers are mentioned, unless the source document already has line numberings (e.g., an issued patent) every printed line is counted including any headings, but not including any page headers.

Claims 1, 3, 4, 13, 17, and 18 stand rejected under 35 U.S.C. § 102(b) as being anticipated by Kobus. Appellant traverses because Kobus does not disclose each and every element of the invention as set forth in the claims.

A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. of California,* 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). See MPEP 2131. Therefore, the presence of a single claim element not disclosed by the prior art reference cited in the rejection is sufficient to overcome an anticipation rejection under 35 U.S.C. § 102. There are many differences between the claimed invention and Kobus; the claim elements mentioned in the following discussion are presented as examples only of the many differences wherein each difference is sufficient to overcome the anticipation rejection.

A. The cited art does not teach the feature of a programmable error correcting circuit on said chip

Claim 1 sets forth a central processor chip having several components. In addition to processor circuitry (claim 1, line 5) and RAM (claim 1, line 7), claim 1 sets forth "a programmable error correcting circuit on said chip" (claim 1, line 6).

The Examiner suggests that the microprocessor on the key device reads on the processor chip set forth in claim 1 (page 3, lines 1-5). Appellant respectfully disagrees. Kobus discloses a security device for a computer comprising a key device that connects to a computer or other device using a serial (RS-232) connection. This is a type of device commonly known today as a "dongle" and is referred to herein as a key device. Kobus' key device includes a microprocessor and ROM-encoded program and data variables (col. 5, lines 1-2 and 10-12). The key device of Kobus communicates with the computer in a complex handshaking protocol to identify the sub-key (which corresponds with specific software) and pass a double-encrypted security key, as illustrated in Figure 3, of Kobus. There is no mention that the key device in any way performs any error correcting function or has error

correcting circuitry on the chip. The key does not correct errors, it merely contains an encrypted security message used by the main software on the computer to which the key device is connected. The software encoded on the ROM does not contain errors and no encryption or error correction of this code is necessary since, "investigation . . . of the ROM encoded program and data variables is not possible" (col. 5, lines 10-12; see also col. 19, lines 7-12).

The Office Action fails to point out how or what part of Kobus reads on an error correcting circuit. Instead, the Office Action summarizes the operation of Kobus (Office Action, page 3, lines 6-18) and then conclusively states that the reference meets the claim limitations (Office Action, page 3, lines 18 to page 4, line 6).

In the Advisory Action, the Examiner states simply, "Applicant's argument that the Kobus reference does not disclose an error correcting circuit is not persuasive because since [*sic*] Kobus discloses that the microprocessor has error correction functionality (Col. 4, lines 5-40 & Col. 11, lines 42-65), which would meet the limitation of an error correction circuit" (page 2, lines 1-3). Appellant has carefully reviewed the indicated portions of Kobus and is at a loss to find any suggestion that the microprocessor disposed in the Kobus reference has "error correction functionality."

The first portion of Kobus identified by the Examiner as showing "error correction functionality" is column 4 lines 5-40. Appellent believes the Examiner may have misconstrued the text beginning with line 22 and ending at line 30. Specifically, the specification reads:

> "Circuitry is provided in the key for recognizing predetermined ones of security message portions in the software program and in response to such recognition for modifying such portions and for supplying the modified portions from the key back to the computer whereupon the computer generates new predetermined encrypted security message portions for storage in the software as replacements for the original security message portions."

Although one can see how the Examiner might misinterpret this sentence as suggesting that computer instructions are sent to the key device, this is clearly not the case as is evident from a review of the section of the Kobus patent containing the detailed description. The security messages sent from the software program to the key and back are not computer instructions containing errors. The security messages are defined in Figure 2 and include a

number of fields containing data. The data sent to the key is not erroneous and therefore no error correction is made. Rather, the key performs calculations on the data and sends back an encrypted reply. The reply must correspond with the expected reply in accordance with the calculations.

This process is explained with reference to the lower half of Figure 4 wherein a security message is received by a master key and broadcast to various slave keys, one of which is designed to recognize the particular security message (blocks 81-86 in Figure 4). As described in column 16, lines 35-54, at the block preceding block 87, the slave key checks to see if the message applies to it or another slave key. If not then no reply is given and the process dies. If it does apply to the slave key, the procedure flows to block 87 wherein the first portion is descrambled using a selected key stored in the key device. The key checks that the serial number of the software program matches the serial number stored in the key (block 88). The second level of decryption takes place in block 89. After second-level decryption, the inner security message is compared with a plain text portion of the security message to ensure the values match. If they don't then an abort code is placed in the return reply. A new key authorization is calculated at block 100 (column 17, line 61 to column 18, line 22). The new key is not "incorporated into the software" but "stored on the disc as the next value of V_1 for subsequent calculations" (col. 18, lines 17-22).

The second portion of Kobus identified by the Examiner as showing "error correction functionality" is column 11, lines 42-65. The only reference to errors in this paragraph is in the sentence beginning at line 45 which reads, "This error code intentionally introduces errors into the program when it is executed to cause erroneous operating results unless valid security checking has nulled the error code." Appellant respectfully points out that "nulling" an "error code" is not the same thing as error correction. The issue is whether Kobus teaches a programmable error correcting circuit on the chip as set forth in claim 1 of the present application. The Office points to a processor in the key device of Kobus. As established above, Kobus does not send any erroneous information to the key device processor and therefore no correction is made by the key device. The errors intentionally made into the software are not corrected. Rather, they are not executed. Correcting an error in a program is not the same thing as not executing the error. Kobus teaches not executing an error and does not teach or suggest correcting an error. Since no error is corrected in Kobus, Kobus cannot meet the limitation of "a programmable error correcting circuit on the chip."

Appellant respectfully submits that there is nothing, even in the summary of Kobus provided in page 3 of the Office Action, which reads on an error correcting circuit. In fact, the Office Action never even points to ***any*** error correcting function of Kobus. The software that runs on the computer disclosed by Kobus includes statements designed to cause an error and end the program, but the error is *never corrected.* Rather, Kobus will execute statements designed to cause an error only if the security message received from the security key is invalid. See Figure 3, function block 63 and associated text at col. 17, lines 54-61. Thus, the error-causing code is never corrected; it is simply never executed when the security code is valid.

Even if the encryption key provided by the key device of Kobus were used to correct errors in the computer software, a proposition with which Appellant strenuously disagrees, it would not be fair to suggest that the "programmable error correcting circuitry" may be so broadly interpreted as to read on the key device of Kobus. Appellant notes that, during examination, the Examiner is obliged to give claim elements the broadest reasonable interpretation to claim elements, and this interpretation must be consistent with the interpretation that those skilled in the art would reach. *In re Hyatt*, 211 F.3d 1367, 1372, 54 USPQ2d 1664, 1667 (Fed. Cir. 2000); *In re Cortright*, 165 F.3d 1353, 1359, 49 USPQ2d 1464, 1468 (Fed. Cir. 1999). The "broadest reasonable interpretation" should be the "plain meaning" or "ordinary and customary interpretation" of the term, which may be evidenced by a variety of sources, *Brookhill-Wilk 1, LLC v. Intuitive Surgical, Inc.*, 334 F.3d 1294, 1298, 67 USPQ2d 1132, 1136 (Fed. Cir. 2003), including: the claims themselves, *Process Control Corp. v. HydReclaim Corp.*, 190 F.3d 1350, 1357, 52 USPQ2d 1029, 1033 (Fed. Cir. 1999); dictionaries and treatises, *Tex. Digital Sys., Inc. v. Telegenix, Inc.*, 308 F.3d 1193, 1202, 64 USPQ2d 1812, 1818 (Fed. Cir. 2002); and the written description, the drawings, and the prosecution history, see, e.g., *DeMarini Sports, Inc. v. Worth, Inc.*, 239 F.3d 1314, 1324, 57 USPQ2d 1889, 1894 (Fed. Cir. 2001). See MPEP 2111.

The phrase, "error correcting circuit," which is on the processor chip itself, as set forth in line 6 of claim 1, would have been understood to be limited to a specialized error correcting circuit, not a the processor as a whole. Furthermore, the phrase, "error correcting circuit" would not have been interpreted to read on a key device that provides a security code to a software program to prevent an erroneous statement from executing. Even if the security code provided by the key device was used to correct an error--again, a proposition with which

Appellant would disagree--the *key device* is not correcting the error and therefore no circuit in the key device is "error correcting." Such an interpretation would be inconsistent with the plain meaning, the ordinary and customary meaning that persons skilled in the art would attribute to it, and it would make no sense in the context of the claim as a whole, and would be inconsistent with the specification.

Therefore, there is no disclosure by Kobus of a key device having microprocessor with an on-chip programmable error correcting circuit. Since Claim 1 sets forth, "a programmable error correction circuit on said chip," Appellant respectfully submits that claim 1 is not anticipated by Kobus. Furthermore, investigation of other prior art of record reveals that claim 1 is not anticipated or made obvious by any prior art of record. Accordingly, Appellant respectfully submits that claim 1 is allowable, and further, that claims 3, 4, and 13, which depend from claim 1, are also allowable for at least the same reasons as claim 1. Furthermore, claim 17 sets forth, "a programmable error correcting circuit on said chip" (line 6) and claim 18 sets forth, inter alia, "on said chip, correcting said instructions using said error correction control information" (lines 7-8). Accordingly, Appellant respectfully submits that claims 17 and 18, which include limitations similar to that of claim 1 discussed above, should also have been allowed for the same reasons as claim 1, also discussed above.

B. The cited art does not teach the feature of receiving error correcting information and processor instructions containing errors

Claim 1 furthermore sets forth that the error correcting circuit (established above as being located *on the chip*) "receives said error correcting information and processor instructions containing errors" (lines 9-10). The Office Action has identified the key device as containing the processor chip having error correction circuitry above. However, the key device *never* receives processor instructions containing errors as set forth in claim 1. Processor instructions are never sent to the key device, and the software executed by the microprocessor on the key device does not contain errors and is not encrypted. In the Advisory Action, page 2, lines 4-6, the Examiner points to Kobus, column 4, lines 5-40 and col. 11, lines 42-65 as showing an error correcting circuit receiving processor instructions containing errors. Appellant has carefully reviewed the indicated portions of the reference for a suggestion that the key device receives processor instructions containing errors, but could not find such a suggestion.

Rather, Kobus is very clear that only security messages containing specific codes, keys, and other data is transmitted to the key device, and no mention is made of processor instructions being sent to the key device from the computer. The entire security message is mapped out in Figure 2 of Kobus and explained in column 7, line 48 to column 10 line 68. At no point does Kobus suggest that processor instructions containing errors are transmitted to the key device.

Furthermore, Kobus states that the key device includes a chip-based microprocessor with ROM memory (col. 5, lines 2-3), that the ROM contains the software that drives the microprocessor (col. 5, lines 9-12), and that the ROM is secure and cannot be accessed "by the outside world" (col. 5, lines 9-12; col. 19, lines 7-12) which therefore teaches away from encrypting the ROM software or encoding it with errors. This is because there would be no incentive or reason to encrypt or introduce correctable errors into the ROM software if the ROM software is inaccessible to the outside world.

Is the Office interpreting the term, "processor instructions" so broadly as to encompass encrypted and non-encrypted security messages passed back and forth between the key device and the computer? If so, this is an unfair and unreasonable interpretation of the term, "processor instructions" which has plain meaning in the art as being information usable directly by a processor that causes the processor to perform designated operations.

During examination, the claims must be interpreted as broadly as their terms reasonably allow. *In re American Academy of Science Tech Center*, 2004 WL 1067528 (Fed. Cir. May 13, 2004). This means that the words of the claim must be given their plain meaning unless applicant has provided a clear definition in the specification. *In re Zletz*, 893 F.2d 319, 321, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989); Chef America, Inc. v. Lamb-Weston, Inc., 358 F.3d 1371, 1372, 69 USPQ2d 1857 (Fed. Cir. 2004). "PLAIN MEANING" REFERS TO THE ORDINARY AND CUSTOMARY MEANING GIVEN TO THE TERM BY THOSE OF ORDINARY SKILL IN THE ART" (MPEP 2111). It is the use of the words in the context of the written description and customarily by those skilled in the relevant art that accurately reflects both the "ordinary" and the "customary" meaning of the terms in the claims. *Ferguson Beauregard/Logic Controls v. Mega Systems*, 350 F.3d 1327, 1338, 69 USPQ2d 1001, 1009 (Fed. Cir. 2003); *ACTV, Inc. v. The Walt Disney Company*, 346 F.3d 1082, 1092, 68 USPQ2d 1516, 1524 (Fed. Cir. 2003); and *E-Pass Technologies, Inc. v. 3Com Corporation*, 343 F.3d 1364, 1368, 67 USPQ2d 1947, 1949 (Fed. Cir. 2003).

The Office is constrained to interpret words to the broadest *reasonable* interpretation, and it is unreasonable to suggest that "instructions" would have been understood by persons of ordinary skill in the art to mean data. Persons of ordinary skill understand that data can be manipulated by a processor but not executed by a processor. Claim 1 sets forth an error correcting circuit that "receives said error correcting information and processor instructions containing errors." However the Kobus only teaches passing data back and forth between the key device and computer of Kobus. Furthermore, the specification's definition of "instructions" as being distinct from "data" (page 22, lines 27-32) constrains the Office's interpretation thereof.

In page 2 of the Advisory Action, lines 7-10, the Examiner suggested that a person having ordinary skill in the art would interpret the term "instruction" as covering the data sent to the key device since "all information sent a [*sic*] received between circuits in a computer . . . is contained in processor instructions. . . ." Appellant respectfully disagrees. The Eaminer's explanation does not comport with the actual functioning of computer processors.

Computer processors generally maintain a separation between instructions and data. Instructions are loaded into an instruction pipeline where they are processed and executed by various execution units in the processor. The instructions are loaded sequentially in accordance with a program counter number, although various jumps to other parts of a program can occur. Data is loaded into processor registers in response to instructions causing the data to be loaded. Then the data is acted upon, e.g., using an arithmetic logic unit, and the results are stored in a data register.

It is simply not the case, as the Examiner suggests, that "all information sent [and] received between circuits in a computer during computer processing operations is contained in processor instructions" (Advisory Action, page 2, lines 7-10). Furthermore, there is absolutely nothing in Kobus that suggests that the data sent to the key device originates in processor instructions. Furthermore, even if there was such as suggestion, it would be irrelevant since the processor instructions themselves are not sent to the key device. Claim 1 sets forth an error correcting circuit that "receives . . . processor instructions containing errors." If the data that is sent to the key device is not executable in a processor, then it is not processor instructions. If it is not processor instructions, then the limitations of claim 1 are not met by Kobus and claim 1 should be allowed.

Since Kobus does not disclose a programmable error correcting circuit that receives error correcting information and processor instructions containing errors, Appellant respectfully submits that Kobus does not anticipate claim 1. Furthermore, claims 3, 4, and 13 depend from claim 1 and are not anticipated by Kobus for at least the same reasons as claim 1. Furthermore, claims 17 and 18 set forth similar limitations in lines 10-12 and 7-8, respectively, and are therefore also not anticipated by Kobus. Furthermore, none of the references of record, either singly or in combination teach or suggest this element. Therefore, Appellant respectfully submits that pending claims 1, 3, 4, 13, 17, and 18 are allowable and reversal of the final Office Action is respectfully requested.

In addition to the claim elements set forth above, independent claims 1, 17, and 18 set forth additional limitations which distinguish the invention from the prior art. For example, claim 1 sets forth RAM on the chip for storing error correcting information, the RAM being in communication with the programmable error correcting circuit, the programmable error correcting circuit generating corrected processor instructions in response to the processor instructions containing errors and the error correcting information, the corrected processor instructions being capable of being executed by the processing circuitry. In contrast, Kobus does not teach storing error correcting information in RAM on the chip or an error correcting circuit that generates corrected processor instructions which are executable in processing circuitry on the chip. Even if the Office is correct in stating that the processor on Kobus' key device corrects errors in the software program executing on the main computer, claim 1 requires that the corrected software be executable on the same processor chip as the error correcting circuit. Independent claims 17 and 18 contain similar limitations that further limit the invention beyond that which is discussed in the Final Office Action.

C. Dependant Claims Further Distinguish

Dependent claims 3, 4, and 13 depend from claim 1 and further define and distinguish the invention from the prior art, thereby giving rise to separate reasons for allowability. For example, claim 3 sets forth that "error correcting information includes a key that enables selection of error correction specific to an error scheme used to generate said errors" (claim 3, lines 1-3). Kobus does not teach an error scheme nor a key that enables selection of a error correction specific to a selected scheme. As another example, claim 13 sets forth that the "instructions provided to said processor include an intentional introduction of errors which are

correctable with error correction algorithms, said correction algorithms pre-selected according to the key" (lines 1-4). Kobus does not suggest that error in the software program are correctable using error correction algorithms.

Since the prior art and Kobus in particular does not teach each and every limitation set forth in the claims, Appellant respectfully submits that the claims should have been allowed and a reverse of the Examiner's final rejection is earnestly solicited.

For all the foregoing reasons, the rejection of claims 1, 3, 4, 13, 17, and 18 under 35 U.S.C. §103(a) as being unpatentable is improper and should be reversed. Accordingly, Appellant respectfully submits that the anticipation rejection is in error, and requests that the Board of Patent Appeals and Interferences reverse this rejection on appeal.

Respectfully submitted,
MARTINE & PENILLA, LLP

Leonard Heyman
Registration No. 40,418

MARTINE PENILLA & GENCARELLA, LLP
710 Lakeway Drive, Suite 200
Sunnyvale, California 94085
PBM Direct Dial: 408-774-6933
Customer Number 25920

VIII. CLAIMS APPENDIX

CLAIMS ON APPEAL

1. A particularly configurable processor for processing error induced computer programs which are selectively operable on said particularly configurable processor, comprising:

a central processing unit chip;

processor circuitry on said chip;

a programmable error correcting circuit on said chip;

RAM on said chip storing error correcting information, said RAM being in communication with said programmable error correcting circuit; and wherein:

the programmable error correcting circuit receives said error correcting information and processor instructions containing errors that are not capable of being executed by said processing circuitry, and

said programmable error correcting circuit generates corrected processor instructions in response to said processor instructions containing errors and said error correcting information, the corrected processor instructions being capable of being executed by said processing circuitry.

3. The processor of claim 1, wherein said error correcting information includes a key that enables selection of error correction specific to an error scheme used to generate said errors.

4. The processor of claim 1, wherein information provided in compiled computer program data in part controls said error correction, thereby providing complementary error correction with a combination of the error correction key and the information provided in the compiled computer program data.

13. The processor of claim 3, wherein instructions provided to said processor include an intentional introduction of errors which are correctable with error correction algorithms, said correction algorithms pre-selected according to the key.

17. A microprocessor for processing computer programs which are selectively operable on selected ones of individual microprocessors, comprising:

an integrated circuit chip;

instruction processing circuits on said chip;

a programmable error correcting circuit on said chip; and

a memory location for storing error correction information, said programmable error correction circuit selecting an error correction scheme based on said error correction information; and

wherein said programmable error correcting circuit receives instructions having errors and said error correction information, and said instruction processing circuits process corrected instructions generated by said programmable error correcting circuit.

18. A method for processing a computer programs on a microprocessors, the method comprising:

intentionally placing errors in the computer program;

loading instructions of said computer program onto instruction registers on a microprocessor chip;

storing error correction control information on said chip;

on said chip, correcting said instructions using said error correction control information; and

executing said instructions on said chip.